Secure Attribute-based Threshold Signature without a Trusted Central Authority

Sun Changxia
Key Laboratory of Network and Information of Education, Xidian University, Xi’an, China
Information and Management Sciences School, Henan Agricultural University, Zhengzhou, China
Email: sunchang_xia@126.com

Ma Wenping
Key Laboratory of Network and Information of Education, Xidian University, Xi’an, China
Email: wp_ma@mail.xidian.edu.cn

Abstract—Currently, in most attribute-based cryptosystem, the central authority that distributes private keys for attributes assigned to the user must be trusted unconditionally otherwise the systems will soon be collapsed. To solve the problem we propose a new attribute-based threshold signature scheme without a trusted central authority. When the number of user’s attributes reaches the threshold he can sign validly. Additionally, the central authority can be distrusted. We prove that the scheme is existentially unforgeable under selective attributes and adaptive chosen-message attack and is secure against collusion attack.

Index Terms—Identity-based, Attribute-based, central authority, CDH problem, collusion attack

I. INTRODUCTION

In order to simplify the key management procedures of the certificate-based public key infrastructures, in 1984 Shamir [1] introduced the concept of identity-based cryptosystem whose public key is an arbitrary string such as E-mail address, ID, IP address, and so on. Attribute-based encryption (ABE) is developed from the identity-based encryption (IBE) [2], the identity of a user is not a unique string but a set of descriptive attributes. In such a system a party will wish to decrypt a document to all users that have a certain set of attributes. For example, in a computer science department of a university, the header might want to encrypt a document to the identity {professor, doctor, academic pacemaker}, any user who has an identity that contains all of these attributes could decrypt the document. Goyal et al. [3] presented a scheme for fine grained access control of encrypted data that each private key represents a formula describing which sets of attributes must appear on the ciphertext in order for this user to decrypt. The advantage of Attribute-based encryption is that a variety of cryptographic operation and dialogue can be easily done with the partial attributes of user instead of the exact identity of user. Attribute-based cryptography acts more flexible and has a much wider range of applications compared with Identity-based cryptography in many situations.

Attribute-based signature (ABS) has also been greatly developed [4-12], publisher of a signature can profess that the signature is associated with a specific set of attributes or access structure[3], and the verifier can confirm whether the signature is signed by the owner with the corresponding attributes or access structure. ABE is the development of IBE, similarly, ABS is the development of Identity-based signature (IBS), moreover, ABS is based on the identity-based threshold signature. The threshold of ABS is $d$ which is the number of the predictive specific set of attributes in the system. As showed in [9], Secret sharing schemes were first introduced by Shamir[13]. In identity-based $(d,n)$ threshold signature systems the secret is shared with multiple users $n$, and $d$ users or more could recover the secret and commonly generate the threshold signature with his own secret share respectively, while less than $d$ users could not gain any information about the secret. Threshold signature is used to ensure the security of long-term and effective secret and solve the problem of concentration of power through Lagrange polynomial interpolation. In the attribute-based threshold scheme proposed by Shahandashti et al. [4], when the intersection between the signer’s attributes and the verifier’s attributes is $d$ or more, the verifier can verify whether the signature is generated by the publisher of a signature. This scheme is a verifier-based threshold signature, however, in the paper the proposal is based on attributes of user threshold signature scheme whose master private key is shared with attributes of user through Lagrange interpolation, and when the number of user’s attributes reaches the threshold $d$ the user can sign validly.

A. Our Contribution

There is an inherent disadvantage in identity-based systems: the problem of private key escrow [14] [15],

Manuscript received July 31, 2011; revised September 22, 2011;
This research was supported in part by the National Natural Science Foundation of China under Grant No. 61072140.
This research was supported in part by Subject Innovation and Recruitment Program for Colleges and Universities of China under Grant No. B08038.
that is, the trusted Private Key Generator (PKG) has all the user's private key and can easily impersonate any user at any time without being detected. It implies that the PKG must be trusted unconditionally otherwise the systems will soon be collapsed. However, it would be difficult to assume the existence of a trusted party in an ad hoc network, where the communication parties are changing frequently. Similarly, ABS is also facing this problem, there is a trusted central authority (CA) [6] [10] who distributes private key for attributes assigned to the user. So, CA can easily forge the signature of any user at any time without being detected. To solve this problem, there is a trusted central authority. The following is the proof of knowledge, there is no attribute-based threshold signature without a trusted central authority. The challenger chooses 1, 2 \in \mathbb{Z}_p . The adversary then attempts to output $g^{ab} \in G_1$.

We say that the $(t, c) - CDH$ assumption holds in $G_1$ if there is no $t$-time adversary has non-negligible advantage $c$ in solving the CDH problem in $G_1$.

C. Lagrange Interpolation

Shamir [7] uses polynomial interpolation to solve the problem of share of the secret. Let $q(x)$ be a $d$-degree polynomial, given $d$ different points $(x_i, q(x_i))$, we can uniquely compute $q(x)$ for any $x \in \mathbb{Z}_p$:

$$q(x) = \sum_{i=1}^{d} q(x_i) \prod_{j \neq i \neq d} (x - x_j) / (x_i - x_j).$$

We define the Lagrange coefficient:

$$\Delta_{i,j}(x) = \prod_{s \neq j}^{d} (x - j) / (i - j),$$

where $S$ is a set composed of any $d$ different points $(x_i, q(x_i))$ and $S \subset \mathbb{Z}_p$ . Then,

$$q(x) = \sum_{i=1}^{d} q(x_i) \Delta_{i,j}(x).$$

In a $(d, n)$ threshold signature systems, if there is less than $d$ different points $(x_i, q(x_i))$ in $S$, we couldn’t get any information about the polynomial $q(x)$.

III. ATTRIBUTE-BASED SIGNATURE

In this section, we show the definition and security model of attribute-based signature. Then, we propose a construction.

A. Syntax

In ABS, there are two entities [9]: a central attribute authority (CA) and users. The central authority is in charge of the issue of attribute private key to users requesting them. In more details, the definition of ABS is described below.

In a generic attribute-based signature scheme there are four algorithms, namely, setup algorithm Setup, private key extraction algorithm Extract, signing algorithm Sign, and verification algorithm Verify. We use $N = \{1, 2, \ldots, n + 1\}$ to denote the universe of possible attributes. The following is a general description of these algorithms.

Setup The Setup algorithm is run by the master entity CA. It is a probabilistic algorithm that takes as input $l$ where $l$ is the security parameter. It outputs a set of public parameters $MPK$ and $MSK$ as a master secret key. The CA publishes $MPK$ and keeps the master secret to itself.

Extract The Private Key Extraction algorithm is run by the central authority on inputs $MSK$ and a set of attributes $\omega$ that the user owns. The CA generates a secret signing key $SK$ for the user.

Sign The signing algorithm is a probabilistic algorithm and is run by a user who wants to sign a message $m$ with his attributes set for signature $S$ where $S \subseteq \omega \subseteq N$ and

random and outputs $(g, g^x, g^y)$. The adversary then attempts to output $g^{ab} \in G_1$.
It is showed that a signer who has at least \( d \) of the attributes in \( \omega \) can validly generate the signature \( \sigma \). In this paper, we simply choose the set \( S \) such that \( |S| \geq d \). That is, when the number of user’s attributes reaches the threshold \( d \) he can sign validly.

\[ S \geq d \]

**Verify** The verification algorithm is a deterministic algorithm and is run by the verifier that takes as input the public parameters \( MPK \), a message signature pair \((m, \sigma)\) and the set for signature \( S \). The verifier checks if \( \sigma \) is a valid signature. If it is valid, the verifier outputs valid and accepts. Otherwise, the verifier outputs invalid and rejects.

\[ \text{Verify}(m, \sigma, S) \]

\[ \text{valid} \rightarrow \text{OK} \]

\[ \text{invalid} \rightarrow \text{NOT} \]

**B. Security Model**

We now discuss the security model of a threshold ABS scheme. We define a weaker notion of security that is selective-set model [3] for proving the security of the attribute-based system under chosen message attack. This model can be seen as analogous to the selective-ID model [17][18][19] used in identity-based encryption (IBE) schemes [20][21][22]. The selective-attribute game is similar to the standard selective-ID model for identity-based encryption and Identity-Based signature [23][24][25][26][27][28] with the exception that the adversary is only allowed to query for secret keys for attributes which have less than \( d \) with the target attributes.

In this paper we consider a weaker notion of security which is selective unforgeability against chosen message and attribute set attacks. More precisely, we define the following game between a challenger \( C \) and the adversary \( F \).

**Initiation Phase** The adversary \( F \) declares the set of attributes \( \omega' \) that he wishes to be challenged upon.

**Setup Phase** The challenger \( C \) chooses a sufficiently large security parameter \( l \) and runs the Setup algorithm. It gives the adversary the resulting system parameters \( MPK \) and keeps the master key \( MSK \) to itself.

**Query Phase** There are three oracles provided to the adversary for query for the following:

1. **Private Key Extraction Oracle:** \( F \) can perform a polynomial bounded number of private key queries for any attribute set \( \psi \) as long as \( |\psi \cap \omega| < d \) for some predefined number \( d \). The challenger \( C \) responds by running Extract and forwards the private key \( SK \) to the adversary.

2. **Signing Oracle:** The adversary can ask for the signature of any attribute set \( \psi \) on any message \( m \) as long as \( |\psi \cap \omega| < d \). The challenger responds by first running Extract to obtain the private key \( SK \) of the set \( \psi \), and then running Sign to obtain a signature \( \sigma \) that is forwarded to the adversary.

3. **Random Oracle:** Since the security in the paper is proved in the random oracle model, another oracle should also be provided to the adversary additionally. \( F \) can make at most \( q_H \) queries to \( H \)-oracle. Given \( m \), the challenger output a random value to the adversary.

**Forgery Phase** \( F \) outputs a signature \( \sigma' \) on messages \( m' \) with respect to attributes set \( \omega' \).

We say that the adversary wins the game, that is, \( \sigma' \) is a valid signature on message \( m' \) with respect to \( \omega' \) if the following hold true:

1. \( F \) runs the verification algorithm and outputs valid and then accepts.
2. \( F \) adversary has not made a private key query on the attributes set \( \omega' \).
3. \( F \) has not made a sign query on \((m', \omega')\).

The advantage of an adversary that it wins in the above game is defined to be

\[ \text{Adv}_F = \Pr[F \text{ succeeds}] \]

If no polynomial adversary has a considerable advantage in the above game, we say that the t-ABS scheme is selectively unforgeable against chosen message and attribute set attacks, or SUF-CMAA-secure for short.

**Definition1.** An adversary \( F \) is said to be a \((\varepsilon, t, q_\psi, q_H)\) - forger of an attribute-based signature scheme if \( F \) has advantage at least \( \varepsilon \) in the above game, runs in time at most \( t \), and makes at most \( q_\psi \) and \( q_H \) extract and sign and random oracle queries, respectively. A scheme is said to be \((\varepsilon, t, q_\psi, q_H)\) - secure if no \((\varepsilon, t, q_\psi, q_H)\)- forger exists.

**IV. OUR ATTRIBUTE-BASED SIGNATURE CONSTRUCTION**

In our construction, the signer could generate a signature with some of its attributes. And, in our system a predefined number \( d \) will be given before setup algorithm and \( d \) is the threshold. And, the signer could generate a signature with its \( d \) attributes. In addition, we present the construction in large universe and use \( N = \{1, 2, ..., n+1\} \) to denote the set of possible attributes.

Let \( G_1, G_2 \) be cyclic groups of prime order \( p \), and let \( e : G_1 \times G_1 \rightarrow G_2 \) denote the bilinear map. Additionally, let \( g \) be a generator in \( G_1 \).

**A. Setup Algorithm**

For CA, first, choose \( y_i \in Z_p \) at random and compute \( g_1 = g^y \). One hash function is also chosen such that \( H: \{0, 1\} \rightarrow G_1 \). Next, choose \( g_2, t_1, ..., t_{n+1} \) uniformly at random from \( G_1 \). Let \( N \) be the set \( \{1, 2, ..., n+1\} \) and we define a function, \( T \), as:

\[ T(x) = g_3^{s^x} g^{h(x)} = g_3^{s^x} \prod_{i=1}^{n+1} t_i^{h_i(x)} \]
The public parameters of the system are
\[ MPK = \{ g_1, g_2, e, p, g, g_1, t_1, \ldots, t_{n1}, H \} \]
and master key is \( \text{MSK} = y_1 \).

B. Extract Algorithm
To prevent CA from forging the signature of a signer, the signer selects a random integer \( y_2 \in Z_p \) and computes \( g_2 = g_1^{y_2} \); then he keeps \( y_2 \) as his long-term private key and publishes \( g_2 \). Next, CA randomly chooses a \( d - 1 \) degree polynomial \( p(x) \) such that \( p(0) = y_1 \). And the signer selects a \( d - 1 \) degree polynomial \( q(x) \) such that \( q(0) = y_2 \). Furthermore, let the attributes set of the signer be \( \omega \) and choose a set \( S \) such that \( S \subseteq \omega \), \( |S| = d \). Last, CA chooses integer \( r_j \in Z_p \) at random for each \( i \in S \) and outputs the private key:
\[ SK = \{ g_3^{p(i)}(i)^{q(i)} T(1)^{y_2}, g^r \}_{i \in S}. \]

C. Sign Algorithm
Suppose that the message to be signed is \( m \) and the signer’s attributes set for signature is \( S \). Using private key \( SK \), the signer randomly chooses integer \( s_i \in Z_p \) for each \( i \in S \) and outputs the signature:
\[ \sigma = \{ s_1, s_2, s_3 \} \]
\[ = \{ g_3^{p(i)(i)^{p(i)} T(1)^{y_2} H(m)^y} g^r \}_{i \in S}. \]

D. Verify Algorithm
Given \( MPK, g_2, S \) and a message signature pair \((m, \sigma)\), the verifier computes \( Z = e(g_2, g_3) \) and verify that:
\[ \prod_{i \in S} \left( e(T(i), \sigma_i) \cdot e(H(m), \sigma_i) \right)^{\frac{q(i)}{t}} = Z. \]

If the equality holds, output valid. Otherwise, output invalid.

V. SECURITY PROOFS

A. Correctness
We show that our scheme is correct. If \( \sigma = \{ s_1, s_2, s_3 \} \) is a correctly produced signature, we have
\[ \prod_{i \in S} \left( e(T(i), \sigma_i) \cdot e(H(m), \sigma_i) \right)^{\frac{q(i)}{t}} = Z. \]

\[ \prod_{i \in S} \left( e(g_3, g_1^{y_2}) \cdot e(H(m), g_1^{y_2}) \right)^{\frac{q(i)}{t}} = Z. \]

B. Unforgeability

Theorem 1 The above scheme is existentially unforgeable under selective attributes and adaptive chosen-message attack if the CDH problem is hard.

Proof We prove the scheme with reduction of the theory of provable security. Suppose that an adversary F has an advantage \( \varepsilon \) in attacking the scheme, we can build an algorithm B that uses F to solve the CDH problem.

The algorithm B will be given a group \( G_1 \), a generator \( g \) and an instance \((g, g^e, g^h)\) of the CDH problem, and then B computes \( g^{ab} \). To be able to use F to compute \( g^{ab} \), B must be able to simulate a challenger C who is service for F. That is to say, the algorithm B is a simulator. The simulation proceeds as follows:

1. **Initiation** The adversary F outputs the set of attributes \( \omega \) that he wishes to be challenged upon.

2. **Simulation of Setup** B sets \( g_3 = g^e = g^{n(y_2)} \) and \( g_1 = g^h \), then computes \( z = (g_2, g_3) \). As in [2], B chooses randomly an \( n \) degree polynomial \( u(x) \) and another \( n \) degree polynomial \( f(x) \) such that \( u(x) = -x^e \) if and only if \( x \in \omega \). B then sets
\[ t_i = g_3^{u(i)(i)^{f(i)}} \text{ for } i = 1, 2, \ldots, n + 1. \]

Then, we implicitly have
\[ T(x) = g^{x+u(x)} g^{f(x)} \]

Since
\[ T(x) = g_3^{x} \prod_{i=1}^{n+1} t_i \]
\[ = g_3^{x} \prod_{i=1}^{n+1} g^{u(i)(i)^{f(i)}} \]
\[ = g_3^{x} \sum_{i=1}^{n+1} u(i)(i)^{f(i)} \]
\[ = g_3^{x} u(x) g^{f(x)} \]

The simulator B gives the public parameters:
\[ MPK = \{ g, g_1, g_2, t_1, \ldots, t_{n1}, H, z = (g_2, g_3) \}, \]
The corresponding master key, $MSK = y_1$, is unknown to B.

**Simulation of Random Oracle** Assume $F$ makes at most $q_d$ times query to $H – \text{oracle}$. The challenger $C$ maintains a list $L$ to store the answers of $H$. Meanwhile, it selects a random integer $\eta \in [1, q_d]$. If $m_i$ is the query of $H$, $B$ checks the list $L$. And it works as follows: If an entry for the query is found in $L$, the same answer will be returned to the query of $F$. Otherwise, $B$ has two choices:

1. If $i \neq \eta$, it chooses two random integers $\alpha_i, \beta_i \in Z_p$ and answers $H(m_i) = g_2^{\alpha_i} g_\beta^\beta$.
2. If $i = \eta$, it chooses a random integer $\beta_i \in Z_p$ and answers $H(m_i) = g_\beta^\beta$.

**Simulation of Private Key Extraction Oracle** $F$ can make requests for private keys on the attributes set $\gamma$ such that $\gamma \cap \omega^* < d$. To answer a private key query on $\gamma$, the simulator $B$ proceeds as follows:

Firstly, we define three sets $\Gamma, \Gamma', S$ in the following manner:

$$\Gamma = \gamma \cap \omega^*, \Gamma' \subseteq \Gamma, |\Gamma'| = d - 1,$$

and $S = \Gamma \cup \{\emptyset\}$.

Then we define the private key $SK_{\gamma}$ for $i \in \Gamma'$ as:

$$SK_{\gamma} = \left\{ g_{3}^{k_{i}} \cdot \Delta_{i}(i), g_{3}^{\gamma} \right\}$$

where $k_{i}, \Delta_{i}, \gamma$ are chosen randomly from $Z_p$.

We define a $d - 1$ degree polynomial $p(x)$ such that $p(0) = y_1, p(i) = k_{i}$ and another $d - 1$ degree polynomial $q(x)$ such that $q(0) = y_1, q(i) = \lambda_i$. It is also noted that random choices of elements $k_{i}, \Delta_{i}$ implies random elements $k_{i}, \lambda_i$.

Next we compute the private key $SK_{i}$ for $i \in \gamma \setminus \Gamma'$ as follows:

$$SK_{i} = \prod_{j \in i} g_{3}^{k_{j} \cdot \Delta_{j}(i)} \left( g_{2}^{\bar{p}(i)} \cdot g_{3}^{\gamma} \right)^{i} \cdot \Delta_{i}(i)$$

So, the private key $SK_{i} = \left(SK_{\gamma}, SK_{i}\right)$ could also be simulated.

Additionally, Since $i \in \gamma \setminus \Gamma'$, $i + u(i)$ will be non-zero. Let $r_i = (r_i^{-1} - \frac{y_1 y_2}{i + u(i)}) \Delta_{i}(i)$, the distribution of the private key $SK_{i}$ for $i \in \gamma \setminus \Gamma'$ is the same as those of the private key for $i \in \Gamma'$. In a word, the private key could be simulated validly by the algorithm $B$.

**Simulation of Signing Oracle** $F$ also makes requests for signature query on attributes $\gamma$ such that $\gamma \cap \omega^* < d$.

If $|\gamma \cap \omega^*| \geq d$, the simulator $B$ aborts. Otherwise, $B$ selects a random set $\Omega$ such that $\Omega \subset \omega^*$ and $|\Omega| = d - 1$.

For these $d - 1$ points of $\Omega$, the signature is computed as:

$$\sigma = \left\{ g_{3}^{k_{j} \cdot \Delta_{j}(i)}, g_{2}^{a_{j}}, g_{\beta}^{\beta} \cdot T(i) \right\}$$

where $k_{j}, \lambda_{j}$ is chosen randomly in $Z_p$ defined for all $i \in \Omega$. It is showed that the signature could also be simulated.

To answer the signature query on $\gamma$, the signature of the $d - th$ point is also be computed by $B$. With Lagrange interpolation and $p(0)q(0) = y_1 y_2$, $B$ simulates the signature as:

$$\sigma_1 = \prod_{i = 1}^{d - 1} g_{3}^{k_{i} \cdot \Delta_{i}(i)} \cdot g_{3}^{a_{i}} \cdot g_{2}^{\beta} \cdot \Delta_{i}(i) \cdot T(i)$$

$$\sigma_2 = g_{2}^{\beta}$$

$$\sigma_3 = g_{3}^{a_{i}} \cdot g_{\beta}^{\beta}$$

$$\sigma = \left\{ \sigma_1, \sigma_2, \sigma_3 \right\}$$

where $s_{d}, r_{d} \in Z_p$ and $H(m) = g_{2}^{a_{d}} g_{\beta}^{\beta}$.

Let $s_{d} = - \frac{\Delta_{i}(i)}{a_{d}} + s_{i + u(i)}$, then we have

$$\sigma = \left\{ g_{3}^{k_{i} \cdot \Delta_{i}(i)} \cdot T(i), \Delta_{i}(i), H(m) \cdot g_{\beta}^{\beta}, g_{2}^{\beta} \right\}$$

It shows that the signature has the correct distribution. For answering the signature query on the attributes set $\gamma$ such that $|\gamma \cap \omega^*| < d$, the simulator $B$ uses $SK_{i}$ to generate a signature on $m$ exactly as in the actual scheme, and outputs the result. Thus, $B$ can correctly simulate the signature.

**Forgery** Finally, the adversary outputs a forged signature on message $m^*$ for the attributes set $\omega^*$:

$$\sigma^* = \left\{ \sigma_{i'}, \sigma_{i''}, \sigma_{i}^* \right\}$$

If $|\gamma \cap \omega^*| \geq d$ or $H(m^*) \neq g_{\beta}^{\beta}$, the simulator $B$ aborts. Otherwise, the following equation holds:

$$\prod_{i \in S} \binom{\epsilon(T(i), \sigma_{i}^*)}{\epsilon(H(m), \sigma_{i}^*)}^{\frac{1}{2x_{i}(i)}} = Z$$

Now, note that $T(i) = g^{x(i)}$ and $H(m) = g_{\beta}^{\beta}$, we have

$$\prod_{i \in S} \binom{\epsilon(g^{x(i)}, \sigma_{i}^*)}{\epsilon(g_{\beta}^{\beta}, \sigma_{i}^*)}^{\frac{1}{2x_{i}(i)}}$$

$$= \epsilon(g^{x}, g_{\beta}^{\beta})$$

Thus, $B$ will compute $g^{x\beta}$ and solve the CDH problem as follows:
\[ g^{ab} = \prod_{i \in S} \left( \frac{\sigma_1^i \ast \sigma_2^{x_i} \sigma_3^{y_i}}{\sigma_2^{x_i} \sigma_3^{y_i}} \right) . \]

C. Security against Collusion Attacks

The definition of collusion attack is that multiple users combine their keys to form identities that are a combination of their attributes. Then, the colluders are then able to sign that none of them individually were able to sign. The construction in the paper can resist collusion attacks like other attribute-based schemes [2-12], because the private key components are tied to random polynomials that CA selects for different users, each user’s attributes keys are generated using different random share of a secret, so keys generated for different users cannot be combined.

D. Security against the Forgery of the Central Authority

In [2-12], there is a critical disadvantage that the central authority knows the private keys of all users, so he is able to impersonate any user to sign a document or decrypt an encrypted message. If the central authority is corrupted the system will soon be collapsed. This is called the problem of private key escrow that exists in ID-based systems and attribute-based systems. In this paper we proposed a threshold signature scheme without trusted CA. To prevent CA from forging signature of a user, CA has master key \( y \), of the system and the user has his long-time private key \( y_2 \), the master key of CA and the private key of the user are shared by the attributes of the user, so the private keys of the attributes are determined by CA and the user. Thus, CA cannot forge signature of the user without knowing \( y_2 \).

VI. PERFORMANCE ANALYSIS

Compared with other attributed-based signature schemes [5][9], our proposal has two advantages: one is that the signer only needs to publish part of his attributes as the set of signature such that \( S \subseteq \omega, |S| = d \) instead of the whole set of attributes \( \omega \). So, the verifier is only aware of threshold \( d \) and the set of signature, and cannot precisely knows all attributes owned by the signer. Thus, the privacy of the signer is well protected. Another is that there are \( d \) private keys for attributes in \( S \) than in \( \omega \) to be generated in the key generation algorithm. If there is a great deal of attributes in \( \omega \), the scheme proposed in this paper can greatly improve in efficiency.

VII. CONCLUSION

An attribute-based threshold signature scheme without a trusted central authority is firstly put forward in this paper whose threshold is based on attributes of signature for the user than those of intersection between the user’s and the verifier’s. When the user has more than \( d \) attributes, he can effectively sign. Additionally, the problem of private key escrow is well solved, and so this scheme increases its applicability. Additionally, our scheme is proved existentially unforgeable against adaptively chosen message attack and secure against collusion attack.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61072140, and Subject Innovation and Recruitment Program for Colleges and Universities under Grant No. B08038.

REFERENCES


---

Sun Changxia is a Ph.D. student at Key Laboratory of Network and Information of Education, Xidian University, Xi’an of Shanxi province in China and a lecture at Information and Management Sciences school, Henan Agricultural University, Zhengzhou of Henan province in China. Her research fields are digital signature and theory of provable security.

Ma Wenping is a tutor of doctoral students and a professor of Xidian University. He is the head of Key Laboratory of Network and Information of Education. His research fields are sequences design of cryptosystem and theory of provable security. He is author of many articles published in international peer reviewed journals.